

I.E. Gabriel García Márquez

Cristian Dagnover Machado

Mateo Pulgarin García

Hawrisson Delvis Avendaño

Grado 10º2

- ¿Qué son estructuras de control?

R//:son los que permiten modificar el flujo de ejecución de las instrucciones de un programa.

Con las estructuras de control se puede:

- De acuerdo a una condición, ejecutar un grupo u otro de sentencias (If-Then-Else y Select-Case)
- Ejecutar un grupo de sentencias mientras exista una condición (Do-While)
- Ejecutar un grupo de sentencias hasta que exista una condición (Do-Útil)
- Ejecutar un grupo de sentencias un número determinado de veces (For-Next)
- Etc.

- **Qué son Estructuras cíclicas o repetitivas?,Cuál es su sintaxis? Ejemplos?**

R//: son aquellos en cuya solución es necesario utilizar un mismo conjunto de acciones que se puedan ejecutar una cantidad específica de veces y se clasifican en :

Ciclos con un Número Determinado de Iteraciones

- (Para): Son aquellos en que el número de iteraciones se conoce antes de ejecutarse el ciclo. La forma de esta estructura es la siguiente:

Pseudocódigo

```
Para <var> ← <exp1> hasta <exp2> paso <exp3> haga
    <Tareas a repetir>
Fin-Para
```

Diagrama de Flujos



- .Dado un valor inicial exp1 asignado a la variable esta se irá aumentando o disminuyendo de acuerdo a la exp3 hasta llegar a la exp2; si se omite el paso, significa que la variable aumentará de uno en uno.

Ciclos con un Número Indeterminado de Iteraciones

Son aquellos en que el numero de iteraciones no se conoce con exactitud, ya que esta dado en función de un dato dentro del programa.

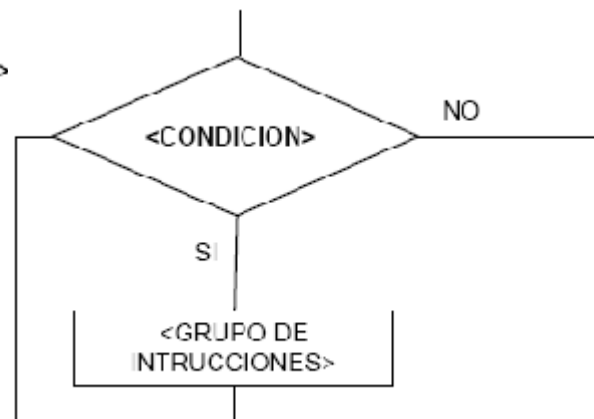
- **Mientras Que:** Esta es una estructura que repetirá un proceso durante $\square N \square$ veces, donde $\square N \square$ puede ser fijo o variable. Para esto, la instrucción se vale de una condición que es la que debe cumplirse para que se siga ejecutando. Cuando la condición ya no se cumple, entonces ya no se ejecuta el proceso. La forma de esta estructura es la siguiente:

Pseudocódigo

```

Mientras Que <condición>
    Accion1
    Accion2
    .
    .
    AccionN
Fin-Mientras
    
```

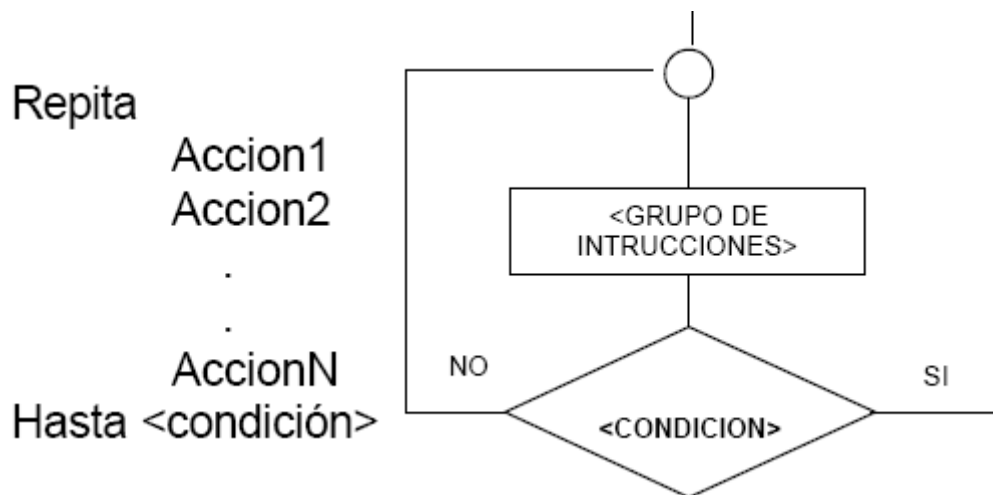
Diagrama de Flujos



- **Repita-Hasta:** Esta es una estructura similar en algunas características, a la anterior. Repite un proceso una cantidad de veces, pero a diferencia del Mientras Que, el Repita-Hasta lo hace hasta que la condición se cumple y no mientras, como en el Mientras Que. Por otra parte, esta estructura permite realizar el proceso cuando menos una vez, ya que la condición se evalúa al final del proceso, mientras que en el Mientras Que puede ser que nunca llegue a entrar si la condición no se cumple desde un principio. La forma de esta estructura es la siguiente:

Pseudocódigo

Diagrama de Flujos

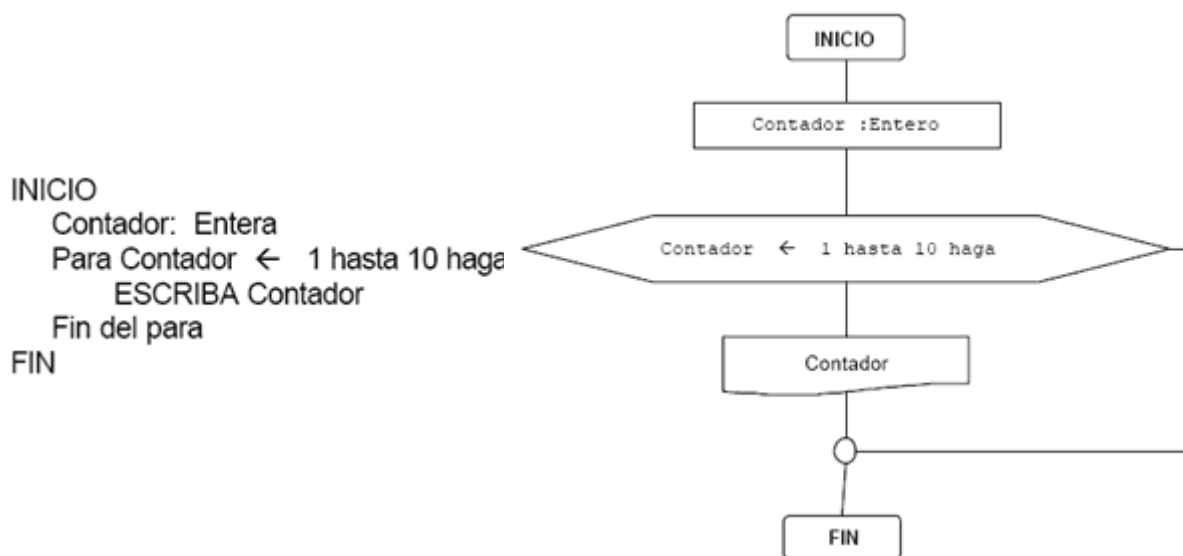


Ejemplo 1:

Realizar un algoritmo que muestre los números de uno en uno hasta diez usando una estructura Para. Exprese el algoritmo usando Pseudocódigo y diagrama de flujos.

Pseudocódigo

Diagrama de Flujos

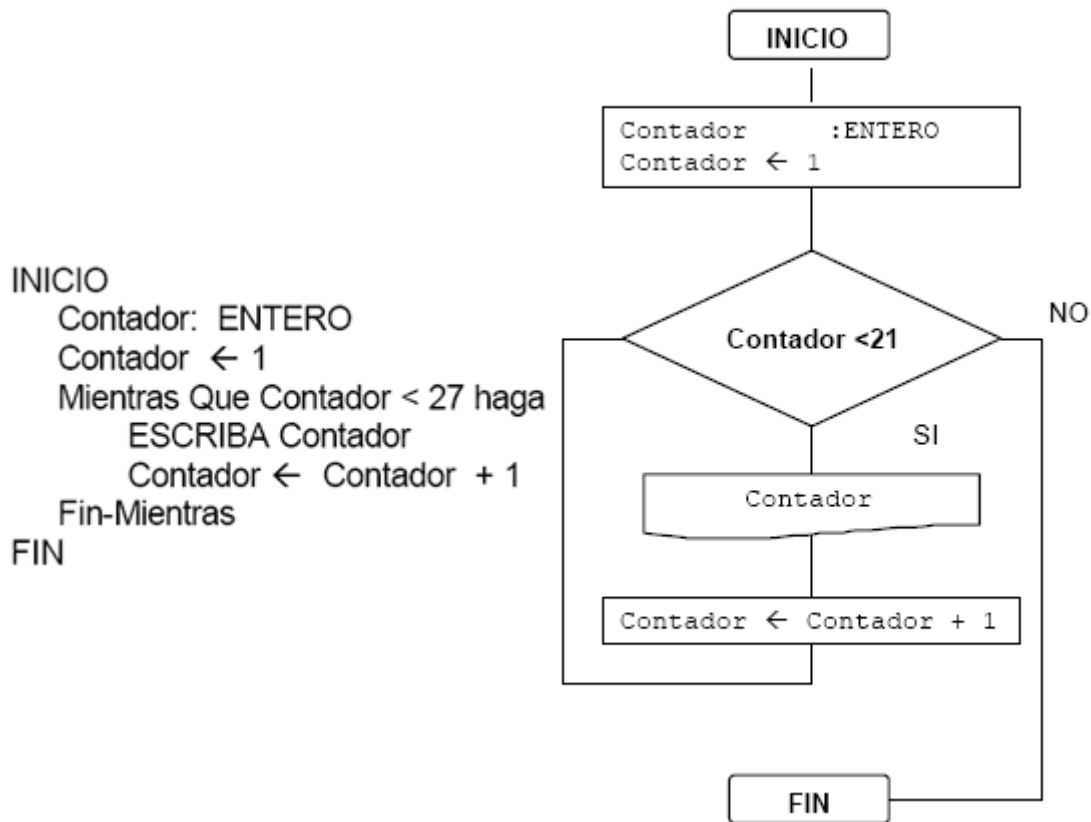


Ejemplo 2:

Usando una estructura Mientras, realizar un algoritmo que escriba los números de uno en uno hasta 20

Pseudocódigo

Diagrama de Flujos



Ejemplo 3:

Realizar un algoritmo que pregunte al usuario un número comprendido en el rango de 1 a 5. El algoritmo deberá validar el número, de manera que no continúe la ejecución del programa mientras no se escriba un número correcto.

INICIO
 Numero : ENTERO
 Escriba "Diga un numero de 1 a 5"
 Repita
 Lea Numero
 Hasta que (Numero >= 1) Y (Numero < 5)
 FIN



- Qué son Esquemas: cuantitativo y cualitativo? Ejemplos?
R//:

El esquema cuantitativo

Es utilizado cuando se conoce el número de veces que debe repetirse un ciclo determinado, antes de activarse la estructura repetitiva. Para el ejemplo resuelto No. 18 antes de llegar al ciclo se proporciona la cantidad de estudiantes, que en este caso sería igual a la cantidad de registros a procesar.



Donde N es el número de registros y su valor es proporcionado al principio del proceso.

EJEMPLO

(Generalización del ejercicio N° 16) Hacer un algoritmo que encuentre la suma de los primeros N números naturales.

Análisis Datos de entrada

- La cantidad de números a tener en cuenta en la suma.

Datos de salida

- La suma de los primeros N números naturales.

Proceso

Primero se debe conocer la cantidad de números naturales a sumar y luego generar y sumar los números comprendidos entre 1 y esa cantidad. El proceso sería el mismo utilizado en el ejercicio 16.

Definición de variables

N: Cantidad de números naturales a sumar. NUM: Contador que genera los números entre 1 y N y que a su vez controla el ciclo. S
UMA: Suma de los números entre 1 y N.

Algoritmo

INICIO SUMA = 0 NUM = 1 LEA: N MIENTRAS NUM

≤

N HAGASUMA = SUMA + NUM NUM = NUM +
1 FIN_MIENTRASESCRIBA: "LA SUMA ES:", SUMAFIN_INICIO

Prueba de escritorio

Si el valor de N es 7: N NUM SUMA 7 1 02 13 3

4 65 106 157 218 28

Salida

Esquema cualitativo

Este esquema es utilizado cuando no se conoce el número de veces que debe repetirse la secuencia de un determinado ciclo. En esta clase de ciclo el usuario decide cuándo terminar con el proceso repetitivo sin importar cuantas iteraciones se hayan efectuado; a diferencia de los ciclos anteriores, en donde el número de iteraciones está determinado desde el principio

Ejemplo

Hacer un algoritmo que encuentre la suma y el promedio de un grupo de datos numéricos positivos que entran de a uno por registro.

Análisis

Datos de entrada

- El valor de cada número

Datos de salida

- La suma de los números
- El promedio del grupo de números.

Proceso

Como los datos entran de a uno por registro, se lee un número, se suma y se vuelve a leer el próximo para sumarlo también. ¿Cuándo terminar este proceso? Como en este caso solamente hay un campo en el registro se deben analizar las características de éste para escoger el valor que no esté permitido para él, de acuerdo a lo que nos dice el enunciado. Ahora bien, el campo es numérico positivo, por lo que ni el valor cero ni los valores negativos serían tenidos en cuenta en la suma de los números. Esto permite escoger un valor para la marca que esté en el rango menor o igual a cero. Si se escoge el valor cero, significa que se adicionará un registro con este valor y que cuando se encuentre es porque no hay más valores a sumar. En otras palabras, se lee un número, se compara con cero, si es mayor que cero se tiene en cuenta para la suma, se lee el próximo y se le hace la misma comparación y el proceso será cíclico, hasta que se encuentre un valor igual a cero; esto hará que se termine el ciclo de lectura. Para encontrar el promedio es necesario contar los números que se van a sumar y su cálculo debe hacerse fuera del ciclo cuando se obtenga la suma total.

Promedio = $\frac{\text{suma}}{N}$; pero este N no está dado por el enunciado, por lo tanto hay que calcularlo.

Definición de variables

NUM: Variable que almacenará uno a uno cada número o valores a sumar, una vez se ejecuta una instrucción de entrada.

SUMA: Acumulador que totalizará los valores a entrar

N: Contador que contabilizará el total de números.

PROM: Promedio de los números.

Algoritmo

INICIO SUMA, N = 0 (1) LEA: NUM (2) MIENTRAS NUM > 0
HAGA SUMA = SUMA + NUM (3) N = N + 1 (4) LEA: NUM (5) FIN_MIENTRAS
PROM = SUMA / N (6) ESCRIBA: "LA SUMA ES:", SUMA "Y EL PROMEDIO", PROM (7) FIN_INICIO

Qué es la Estructura Mientras?,Cuál es su sintaxis? Ejemplos?

Mientras es una estructura algorítmica que se ejecuta mientras la condición evaluada resulte verdadera. Se evalúa la expresión booleana y, si es cierta, se ejecuta la instrucción especificada, llamada el cuerpo del bucle. Entonces se vuelve a evaluar la expresión booleana, y si todavía es cierta se ejecuta de nuevo el cuerpo. Este proceso de evaluación de la expresión booleana y ejecución del cuerpo se repite mientras la expresión sea cierta.

Cuando se hace falsa, finaliza la repetición. En la lección anterior iniciamos con las estructuras repetitivas.

Representación pseudocodificada:

Español	Inglés
Mientras <condición>	While <condición> do
Acciones	Acciones
Fin_mientras	end_while

EJEMPLO:

Calcular la suma de los cuadrados de los primeros 100 números enteros y escribir el resultado.

Solución.

Hagamos el algoritmo:

Inicio

Suma \leftarrow 0

$i \leftarrow 1$

Mientras $i \leq 100$ hacer

Suma \leftarrow suma + $i * i$

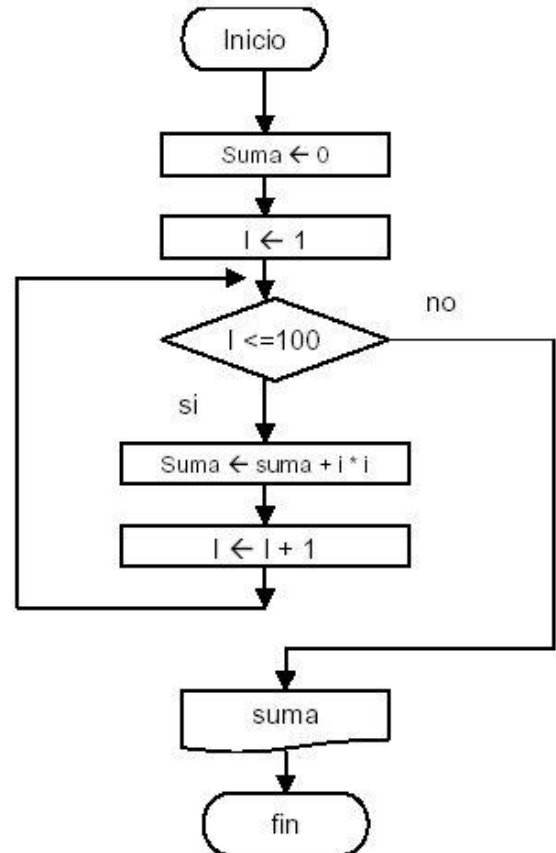
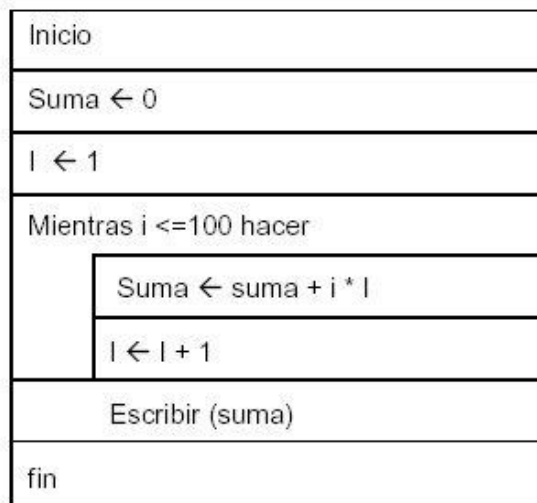
$i \leftarrow i + 1$

Fin_mientras

Escribir (suma)

Fin

Flujograma:



¿Qué es Rompimiento de: ciclos y control de ejecución? Explica por medio de un ejemplo

Dentro de la programación, algunas veces, es necesario hacer que un ciclo se detenga abruptamente (aborte), porque ya se cumplió algo que se necesitaba o se estaba buscando, por lo que, posiblemente, no se alcance a satisfacer completamente en una forma normal la culminación de la cantidad de veces que debe ejecutarse o repetirse un ciclo. La ruptura se hace cambiando el sentido de la expresión lógica que controla el ciclo, para que ésta sea falsa y no se continúe ejecutando la secuencia de instrucciones.

Se tiene una serie de N números enteros. Se quiere saber si dentro de ellos existe, por lo menos, uno cuyo valor sea cero.

Análisis•

La cantidad de números. • El valor de cada número.

Datos de salida

• La salida será un mensaje cuyo contenido dependerá de si se encuentra o no el valor cero dentro de los N números.

Si hay un valor cero, la salida será: "SI HAY UN CERO ENTRE LOS NUMEROS"; y sino la salida será: "NO HAY UN VALOR IGUAL A CERO".

Proceso

Como el enunciado dice que son N números este valor será dado al algoritmo; por lo tanto, se puede implementar un esquema cuantitativo a través de un contador que tome valores entre 1 y la cantidad de números. Como el objetivo del algoritmo es decir si se encuentra, o no, un valor cero dentro del grupo de números, cada vez que se entre un valor se compara con cero, si éste es: • Diferente de cero se entra el próximo valor y se le hace la misma operación. • Si es igual a cero se debe detener la búsqueda, por lo tanto se le cambia de valor a la bandera utilizar. Cuando se termine el ciclo es necesario saber si se encontró, o no, un valor cero; esto lo hacemos mediante una variable tipo bandera cuyo valor inicial será F. y se le cambiará de contenido cuando se encuentre un valor igual a cero.

Definición de variables

NN: Cantidad de números en el grupo.

NUM: Cada valor de un número del grupo.

CONTA: Contador de valores entre 1 y NN.

ENCONTRADO: Variable tipo bandera cuyo valor inicial es F.

Algoritmo

```
INICIO
  LEA: NN
  CONTA = 1
  ENCONTRADO = .F.
  MIENTRAS (CONTA <= NN) ^ (ENCONTRADO =.F.) HAGA
    LEA: NUM
    SI NUM = 0 ENTONCES
      ENCONTRADO =.V.
    SINO
      CONTA = CONTA + 1
    FIN_SI
  FIN_MIENTRAS
  SI ENCONTRADO = .V. ENTONCES
    ESCRIBA: "SI HAY UN CERO ENTRE LOS NUMEROS"
  SINO
    ESCRIBA: "NO HAY UN CERO ENTRE LOS NUMEROS"
  FIN_SI
FIN_INICIO
```

Qué son Estructuras Para,Cuál es su sintaxis? Ejemplos?

- Son aquellos en que el número de iteraciones se conoce antes de ejecutarse el ciclo. La forma de esta estructura es la siguiente:

Pseudocódigo

```
Para <var> ← <exp1> hasta <exp2> paso <exp3> haga  
    <Tareas a repetir>  
Fin-Para
```

Diagrama de Flujos



Qué es Estructura Hacer mientras que?,Cuál es su sintaxis? Ejemplos?

- Esta es una estructura que repetirá un proceso durante ∞ veces, donde ∞ puede ser fijo o variable. Para esto, la instrucción se vale de una condición que es la que debe cumplirse para que se siga ejecutando. Cuando la condición ya no se cumple, entonces ya no se ejecuta el proceso. La forma de esta estructura es la siguiente:

Pseudocódigo

```
Mientras Que <condición>  
    Accion1  
    Accion2  
    .  
    .  
    AccionN  
Fin-Mientras
```

Diagrama de Flujos

